

## REMARKS

### STATUS OF CLAIMS

Claims 1-3, 8, 9, 14, 15, and 19-31 are in the case. No claims are amended, cancelled, or added.

### CLAIM REJECTIONS – 35 U.S.C. § 102(e)

The Final Office Action rejected Claims 1-3, 8, 9, 14, 15, and 19-31 under 35 U.S.C. § 102(e) as anticipated by Hopwood et al., U.S. Patent No. 6,223,343 B1 (“Hopwood”). The Applicants respectfully traverse the rejection.

#### Claims 1-3, 8, 9, and 19-31

Claim 1 recites a release control method for providing early development releases of a software system. The method features the steps of providing an early development branch that is designated for incorporation of one or more software modules that provide support for new features and platforms; **receiving, from a plurality of integration units, a plurality of pre-tested software modules**, wherein each of the pre-tested software modules comprises one or more new features or supports one or more new platforms; **committing the pre-tested software modules into the early development branch**; and, using the early development branch, generating a new early development release that contains pre-tested software modules for new features and platforms. Claim 8 recites a system that comprises logic for performing analogous steps. Claim 23 recites a computer-readable medium that carries instructions for causing one or more processors to perform analogous steps.

Hopwood does not teach or suggest committing a **plurality** of pre-tested software modules into a **single** early development branch. Instead, Hopwood discloses, in col. 20, lines 16-66, that each element is associated with its own separate trunk. Hopwood defines a trunk as a

main line of development for **an** (i.e., one) element (see col. 20, lines 22-23; and col. 29, lines 26-31). Hopwood defines an element as a software entity (see col. 25, lines 33-40). Thus, each separate software entity is associated with a separate trunk.

Hopwood defines a branch as “a separate line of development consisting of one or more revisions that diverge from the trunk” (col. 20, lines 25-27). The trunk cannot diverge from itself. Therefore, the trunk itself is not a development branch. Furthermore, by definition, each development branch diverges from the trunk, and not from any other development branch. When a development branch is issued to production, that development branch becomes the trunk (see col. 21, lines 1-2).

For example, Hopwood describes, from col. 20, line 37 to col. 21, line 6, and in FIGs. 13-19, that a trunk is created for a software entity “MCFUN.TXT.” Branches diverging from the trunk represent revisions of “MCFUN.TXT,” and not any other software entities. Checking out “MCFUN.TXT” from a version manager causes a new development branch for “MCFUN.TXT” to be created (see col. 20, lines 41-44). After “MCFUN.TXT” has been edited and checked back in to the version manager, further development of “MCFUN.TXT” requires that “MCFUN.TXT” be checked out again. Checking out “MCFUN.TXT” again causes a new development branch to be created. Each new development branch diverges off of the trunk for “MCFUN.TXT” (see col. 20, lines 53-65). Only one element is associated with a given development branch. Therefore, Hopwood does not teach or suggest committing a **plurality** of elements to a single development branch.

The Final Office Action asserts that developers 90, 96, and 104 depicted in Hopwood’s FIG. 6 are equivalent to early development branches. However, Hopwood actually states that a developer is anyone that develops elements that need to be implemented on the distributed computer system (see col. 15, lines 20-22). Thus, Hopwood discloses that developers are people. Again, Hopwood defines a branch as “a separate line of development consisting of one or more

revisions that diverge from the trunk.” People are not lines of development. Therefore, developers 90, 96, and 104 are not early development branches.

The Final Office action also asserts that Hopwood’s build issuance component (BLDISS) is equivalent to an early development branch. However, Hopwood actually states that BLDISS is a component of a developer platform (see col. 2, lines 14-16). As discussed above, Hopwood defines a branch as “a separate line of development consisting of one or more revisions that diverge from the trunk.” A component of a developer platform is not a line of development. Therefore, BLDISS is not an early development branch.

The Final Office action asserts that Hopwood’s issuance processor commits pre-tested software modules into an early development branch. However, Hopwood actually states that the issuance processor transfers a program and all of its components from a test environment to a production environment (col. 3, lines 59-62). Hopwood also states that when a revision is issued to production, the revision becomes the trunk (col. 21, lines 1-2). Therefore, when the issuance processor transfers a program to a production environment, the issuance processor actually causes a development branch for the program to become the trunk for the program. Again, Hopwood defines a branch as “a separate line of development consisting of one or more revisions that diverge from the trunk.” Because a branch must diverge from a trunk according to Hopwood’s definition, and because the trunk cannot diverge from itself, it follows that a trunk is not a branch. Therefore, in transferring a program to a production environment, Hopwood’s issuance processor causes the program, which is associated with a development branch, to become the trunk, which is not an early development branch.

Each of Claims 2, 3, 9, 19-22, and 24-31, depends, directly or indirectly, from either Claim 1, 8, or 23, and therefore each of Dependent Claims 2, 3, 9, 19-22, and 24-31 includes each of the distinguishing features, described above, of an independent claim from which the dependent claim depends.

A claim is anticipated under 35 U.S.C. § 102(e) only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference. *Connell v. Sears, Roebuck & Co.*, 722 F.2d 1542, 1548, 220 USPQ 193, 198 (Fed. Cir. 1983). Since Claims 1-3, 8, 9, and 19-31 each include at least one limitation not found in Hopwood, namely, “committing the [plurality of] pre-tested software modules for new features and platforms into the early development branch,” Claims 1-3, 8, 9, and 19-31 are patentable over Hopwood under 35 U.S.C. § 102(e).

#### Claims 14 and 15

Claim 14 recites a produce release method for controlling the release of software system code based on a fixed frequency. The method features the steps of **selecting one or more features for inclusion in a new release of the software system code base, wherein a quantity of features selected will allow a next scheduled release of the software system code base to be completed at a required time**; testing the quantity of features selected in a plurality of business units; providing the quantity of features selected to a pre-integration branch of the software system code base only when testing in the business unit is successful; testing the quantity of features selected in the pre-integration branch; and **providing the quantity of features selected to a development branch only when testing in the business units is successful and in time to allow the next scheduled release of the software system code base to be completed in the required time**.

Hopwood does not teach or suggest selecting one or more features for inclusion in a new release of the software system code base, wherein a quantity of features selected will allow a next scheduled release of the software system code base to be completed at a required time. Hopwood also does not teach or suggest providing the quantity of features selected to a development branch only when testing in the business units is successful and in time to allow the next scheduled release of the software system code base to be completed in the required time.

The Final Office Action asserts that the inherent purpose of all testing is to enable a release and commitment decision. However, Hopwood does not expressly or implicitly disclose that that the inherent purpose of all testing is to enable a release and commitment decision. In fact, the Office Action does not allege that Hopwood discloses this. Apparently, the Office Action relies upon supposed common knowledge to support this assertion.

According to MPEP 2144.03, there must be some form of evidence in the record to support an assertion of common knowledge. If the Examiner is relying upon personal knowledge to support this assertion, then the Applicants request that the Examiner provide an affidavit or declaration setting forth specific, factual statements and explanation to support the assertion. The Office Action provides no evidence for the assertion that the inherent purpose of all testing is to enable a release and commitment decision.

In accordance with MPEP 2144.03, the Applicants traverse this assertion. The Applicants challenge the assertion as not properly being officially noticed, and as not properly being based upon common knowledge.

Even assuming for sake of argument that this assertion is true, it does not follow from this assertion that features to be included in a next software release are selected so that the next software release will be completed at a scheduled release time. In the absence of any evidence to the contrary, it is as likely to be true that it is common for a release time to be scheduled so that selected features may be included in the release. Hopwood does not teach or disclose a selection of features based on a scheduled release time.

It also does not follow from the above assertion that features so selected are **only** provided to a development **branch** when testing in the business units is successful and in time to allow the next scheduled release of the software system code base to be completed in the required time. Hopwood discloses that after an element has been completely tested and ready for issuance to production, the revision becomes the **trunk** that is issued to production. However, Hopwood says nothing of any time-related condition that must be satisfied before features are

provided to a development **branch**. Apparently, features may be provided to Hopwood's development **branch** at **any** time, **regardless** of whether testing is successful, and **regardless** of any scheduled release time (see col. 20, line 41, through col. 21, line 2). As discussed extensively above, Hopwood distinguishes between a trunk and a branch. Hopwood actually discloses an example wherein an element is ready for production **after** the element has already been checked in to development branches five times (col. 20, lines 66-67). Therefore, according to Hopwood, an element does not need to be successfully tested when the element is provided to a development branch.

Claim 15 depends from Claim 14, and therefore Claim 15 includes each of the distinguishing features, described above, of Claim 14.

A claim is anticipated under 35 U.S.C. § 102(e) only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference. *Connell v. Sears, Roebuck & Co.*, 722 F.2d 1542, 1548, 220 USPQ 193, 198 (Fed. Cir. 1983). Since Claims 14 and 15 each include at least one limitation not found in Hopwood, namely, "selecting one or more features for inclusion in a new release of the software system code base, wherein a quantity of features selected will allow a next scheduled release of the software system code base to be completed at a required time," and "providing the quantity of features selected to a development branch only when testing in the business units is successful and in time to allow the next scheduled release of the software system code base to be completed in the required time," Claims 14 and 15 are patentable over Hopwood under 35 U.S.C. § 102(e).

MISCELLANEOUS

The Applicants believe that all issues raised in the Final Office Action have been addressed and that allowance of the pending claims is appropriate. The Applicants respectfully request allowance of the pending claims.

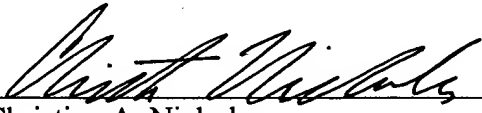
The Examiner is invited to telephone the undersigned at (408) 414-1224 to discuss any issue that may advance prosecution.

To the extent necessary, Applicants petition for an extension of time under 37 C.F.R. § 1.136. The Commissioner is authorized to charge any fee that may be due in connection with this Reply to our Deposit Account No. 50-1302.

Respectfully submitted,

HICKMAN PALERMO TRUONG & BECKER LLP

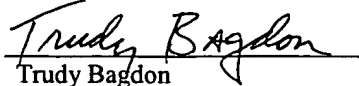
Dated: August 11, 2003

By:   
Christian A. Nicholes  
Registration. No. 50,266

1600 Willow Street  
San Jose, California 95125-5106  
Telephone No.: (408) 414-1080  
Facsimile No.: (408) 414-1076

**CERTIFICATE OF MAILING**

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Mail Stop AF, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450

on 8/11/03 by   
Trudy Bagdon